Report from Tracking Mini-Workshop: Software Status and Future Directions

Alfred Lee
Duke University
Offline Reconstruction Meeting
14 January 1998

outline:

- Report of 16 December meeting.
- Current status.
- Near term goals and effort.
- Long term goals await the appointment of a reconstruction manager ;-)

People whose work is presented here:

Physicists	Peter Berge	FNAL Scientist
	Ken Bloom	Hopkins Postdoc
	Joe Boudreau	Pittsburgh Faculty
	Fritz DeJongh	FNAL Scientist
	Dave Gerdes	Hopkins Faculty
	Chris Greene	Purdue Postdoc
	Betsy Hafen	MIT Scientist
	Alfred Lee	Duke Faculty
	George Pope	Pittsburgh Postdoc
	Alberto Ribon	FNAL Scientist
	Rick Snider	Hopkins Scientist
	Peter Tamburello	Duke Postdoc
	other people	I have forgotten to list

Software Status as of 16 December 1997:

Status of tracking directories:

Tracking and TrackingMods:

- classes: ~200
- files: 459
- C++ source lines: 25,415

Comparison: run I directories:

TRK and SVX:

- include files not counted
- files: 560 (~number of subroutines)
- Fortran source lines: 67,997

…but those lines are debugged!

L3 connection:

Following the agenda…

- First presentation by Kirsten Tollefson: needs of Level III.
- Realism concerning what L3 must do, and cpu time is available (not much! ~1 R10000 CPU-sec/evt).
- First ideas of what sorts of hooks L3 will need in tracking. No big problems seen.
- Current Status: Good ongoing communication between L3 (MIT) and tracking groups.

Silicon Tracking Software:

- Joe Boudreau* presented the status of stand alone 3D pattern recognition in the silicon trackers.
- Several strategies (different seed layers, hit types, etc.) applied in sequence to maximize yield.
- Speed reasonable (right order of magnitude) and efficiency O.K. (~90% on simple events, less on complicated ones).
- Current Status: Ready for "beta." Could also attach SVXII/ISL hits to COT tracks current software.

*Chris Green very involved also.

Insert si_min_bias.ps:
simple min bias event with fits.

Insert si_b_bbar.ps:
complicated $b\bar{b}$ event with fits.

Rick:

- Rick Snider presented briefly his work on vertex banks and fast Z vertex finding in general.
- Like many, Rick splits time with hardware (SI), or teaching, or Run I physics, or other software projects.

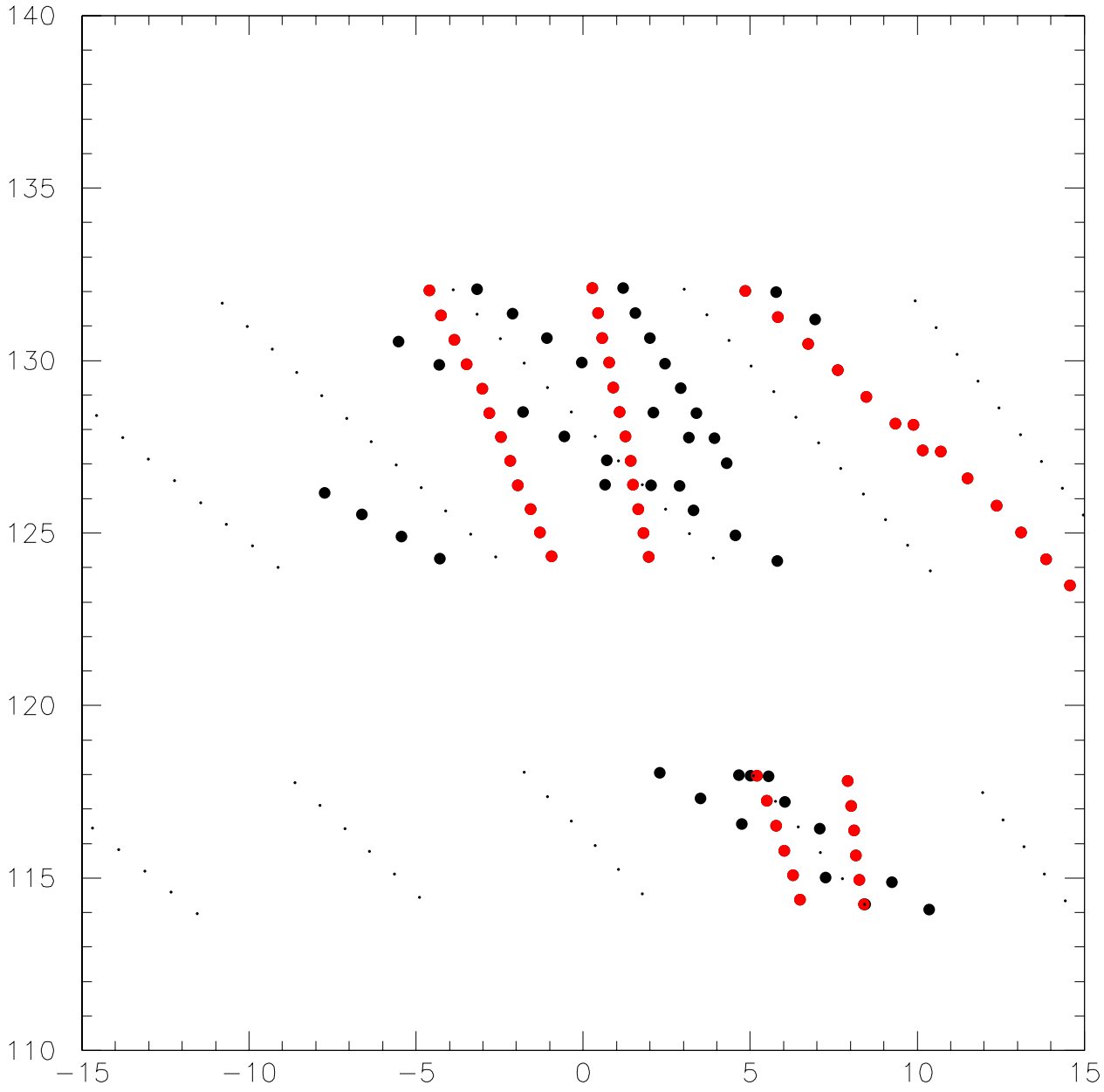
⇒ We must be able to use labor effectively when people have it to give. Three month slices are the best we will get at this time. Open code philosophy makes this easier.

Tracking ideas:

- Pasha Murat presented ideas for improving COT pattern rec. Has prototype system in Root for run I analysis of CTC.
- Betsy Hafen has also presented some similar ideas at tracking meetings concerning "longest list of least conflicting multiples" as a method for "global" pattern recognition.
- As the run II tracking software matures, integrating, with minimum pain, these ideas as alternatives will be a major test of the modularity of the software.

Ken B. and Peter T.

- Ken Bloom and Peter Tamburello presented work on COT/CTC pattern recognition.
- Rapid progress was made between 1 November (when Ken and Dave Gerdes joined the effort), and the end of December.
- Pattern recognition is >80% on simple COT MC, and respectable on CTC.
- Current Status: Pursuing detailed comparisons with Peter B. and Aseet pattern rec on CTC data. Peter T. has been reading the existing Fortran as well as notes.



Red dots are found segments in CTC data.

Insert track_fits_runII.ps:
COT MC with hits on segments (red), linked segments (blue) and fits (black).

Insert track_fits.ps:
CTC data with hits on segments (red), linked segments (blue), and 2D fits (black).

Other issues:

- Al Lee outlined a method for calling C++ object member functions from within Fortran. Allows access to, e.g., geometry objects from within existing wrapped Fortran.
- Also outlined plans for "Modular" or "Object" wrapping the old Tracking code. Open questions are level of wrap, and what about "useless" detectors.
- Current Status: Awaiting Reconstruction Manager for global plan. ;-) Dave Gerdes is looking at wrapping Aseet's fitting routines, as a first step.

Progress since December:

- Progress since December has been along same lines.
- Ken B. almost has Peter T's code working in **AC** as CTCPR3 with existing code. Allow using existing tools to study efficiency in detail.

⇒ Does anyone know how to call cout << in a C++ routine within a Fortran main program?

- Joe B. and Chris Green have continued to improve (speed, accuracy, and durability) the SI stand alone pattern recognition code.

Future Directions

Future directions for the next month or so seem to be the following:

- As soon as MC has a stable release, create a stable Tracking AC++ module that can process run II MC events.

Technical issues:

- Get geometry from Pasha's header files (Si; COT already does this).
- Make a combined COT/SI AC++ module: do we want stand alone SI tracking or putting SI hits on COT tracks? Maybe both.

More Future Directions:

- With AC++ module, should be easier for others (Karlsruhe, Rutgers, Pasha, Betsy) to try algorithm ideas for both COT and SI.
- Wrapping fitting routines will give a good idea of requirements for a full wrap of existing tracking code.
- Additional effort will be needed; any volunteers?

⇒ Despite (endless) turmoil, not too far behind schedule from November. See TrackingDocs/milestones.txt for more.